

# Example 7.6 Scalar first passage times

Etienne A.D. Pienaar

2015

```
library(fptdApprox)
# Under `fptdApprox`:
# Define the diffusion process and give its transitional density:
OU <- diffproc(c("alpha*x + beta","sigma^2",
"dnorm((x-(y*exp(alpha*(t-s)) - beta*(1 - exp(alpha*(t-s)))/alpha))/
(sigma*sqrt((exp(2*alpha*(t-s)) - 1)/(2*alpha))),0,1)/
(sigma*sqrt((exp(2*alpha*(t-s)) - 1)/(2*alpha)))",
"pnorm(x, y*exp(alpha*(t-s)) - beta*(1 - exp(alpha*(t-s)))/alpha,
sigma*sqrt((exp(2*alpha*(t-s)) - 1)/(2*alpha)))"))
# Approximate the first passgage time density for OU, starting in  $X_0 = 3$ 
# passing through  $5+0.25*\sin(2*\pi*t)$  on the time interval  $[0,10]$ :
res1 <- Approx.fpt.density(OU, 0, 10, 3,
"5+0.25*sin(2*pi*t)", list(alpha=-0.5,beta=0.5*5,sigma=1))

library(DiffusionRgqd)
# Under `DiffusionRgqd`:
# Define the diffusion process
G0=function(t){0.5*5-0.5*pi*cos(2*pi*t)-0.5*0.25*sin(2*pi*t)}
G1=function(t){-0.5}
Q0=function(t){1}
# Approximate the first passgage time density for OU, starting in  $X_0 = 3$ 
# passing through  $5+0.25*\sin(2*\pi*t)$  on the time interval  $[0,10]$ :
res2 <- GQD.TIpassage(Xs=3,B=5,s=0,t=10,delt=1/200)

# Let's compare the resulting densities:
plot(res1$y-res1$x,type='l',col='red',xlab='Time(t)',ylab='FPT Density',
main= 'DiffusionRgqd vs. fptdApproximate.')
lines(res2$density-res2$time,col='blue',lwd=2,lty='dashed')
legend('topright',lty=c('solid','dashed'),col=c('red','blue'),legend=
c('fptdApproximate','DiffusionRgqd'),lwd=c(1,2),bty='n')
axis(2,at=seq(0,0.6,by=1/10/10),tcl=-0.2,labels=NA)
axis(1,at=seq(0,10,by=1/4),tcl=-0.2,labels=NA)

# Simulate the first passage time density:
theta <- 0.5
mu <- function(X,t){theta[1]*(10+0.2*sin(2*pi*t)+0.3*sqrt(t)*(1+cos(3*pi*t)))*X-theta[1]*X^2}
sigma <- function(X,t){sqrt(0.1)*X}
simulate.fpt=function(N,S,B,delt)
{
X=rep(S,N)
Ndim=N
time.vector=rep(0,N)
t=1
k1=0
k2=0
pb=txtProgressBar(min=0,max=Ndim,initial=0,char = "=",width = 59, style = 3)
```

```

while(Ndim>=1)
{
k1=k1+1
setTxtProgressBar(pb,N-Ndim)
X=X+mu(X,t)*delt+sigma(X,t)*rnorm(Ndim,sd=sqrt(delt))
# Check if the barrier is crossed and keep trajectories that
# have survived:
I0=X<B
X=X[I0]
count=sum(!I0)
Ndim=length(X)
t=t+delt
if(count>0)
{
time.vector[k2+1:count]=t
k2=k2+count
}
}
close(pb)
return(time.vector)
}

res.sim <- simulate.fpt(500000,8,12,1/2000)

GQD.remove()
# Redefine the coefficients with a parameter theta:
G1 <- function(t){theta[1]*(10+0.2*sin(2*pi*t)+0.3*prod(sqrt(t),1+cos(3*pi*t)))}
G2 <- function(t){-theta[1]}
Q2 <- function(t){0.1}
# Now just give a value for the parameter in the standard fashion:
res3=GQD.TIpassage(8,12,1,4,1/100,theta=c(0.5))

# Compare the numerical solution to the simulated density:
hst=hist(res.sim,freq=F,plot=T,breaks=100)
plot(res3$density~res3$time,type='l',col=2,ylim=c(0,1.0),
main='First Passage Time Density',ylab='Density',xlab='Time',cex.main=0.95)
lines(hst$density~c(hst$mids-diff(hst$mids)[1]/2),type='s',lty='solid',lwd=2)
# Change the parameter and see the effect on the f.p.t. density.
th.seq=seq(0.1,0.5,1/20)
for(i in 2:length(th.seq))
{
res3=GQD.TIpassage(8,12,1,4,1/100,,theta=c(th.seq[i]))
lines(res3$density~res3$time,type='l',col=rainbow(10)[i])
}
lines(res3$density~res3$time,type='l',col=rainbow(10)[i],lwd=2)
legend('topright',legend=th.seq,col=rainbow(10),lty='solid',cex=0.75,
title=expression(theta[1]))

```